

## The Phase Vocoder

- Phase Vocoder는 점점 더 인기 있는 사운드 분석 도구로 떠오르고 있음

### Phase Vocoder Parameters

- 사용자가 선택한 설정에 따라 달라짐
  - 소리 분석의 성격에 따라 조정되어야 함
1. Frame Size : 들어오는 샘플의 수를 한 번에 분석
  2. Window Type : 표준 타입 중에서 윈도우 타입 선택
  3. FFT Size : 실제 숫자의 샘플 FFT 분석을 제공(보통 1024)
  4. Hop Size or overlap factor : 분석 구간 사이의 시간차이

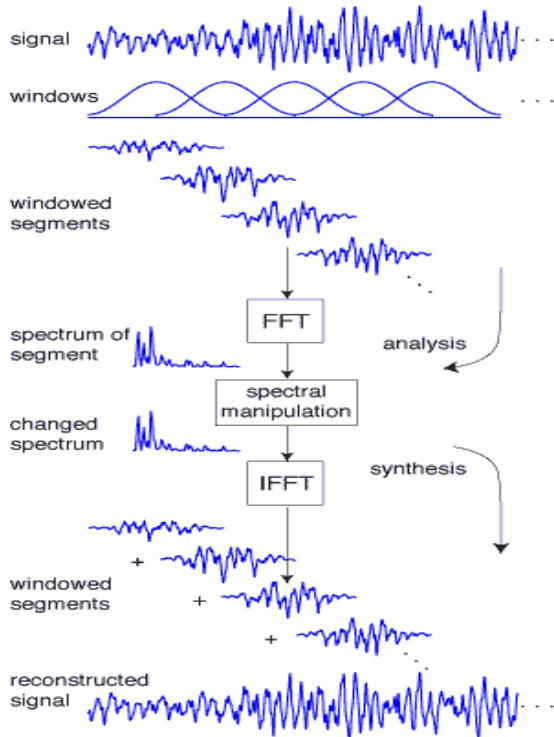
### Frame Size

- Frame Size가 중요한 2가지의 이유
- ① Frame Size의 결정 기준
    - : frequency vs time resolution
  - ② 큰 FFTS는 작은 FFTS 보다 시간이 지연  
real-time system에 부담

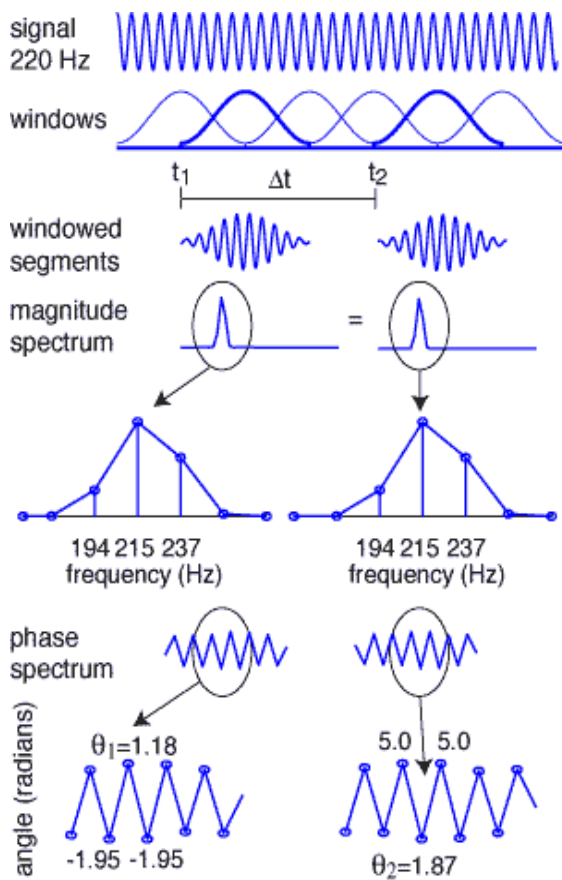
### Window Type

- 대부분 PVs 은 window types 집합체의 옵션을 제공
- hamming / hanning / truncated / gaussian / Blackman-Harris / Kaiser
- window types의 선택은 분석을 하는데 중요

# Spectrum Analysis\_The Phase Vocoder



< STFT



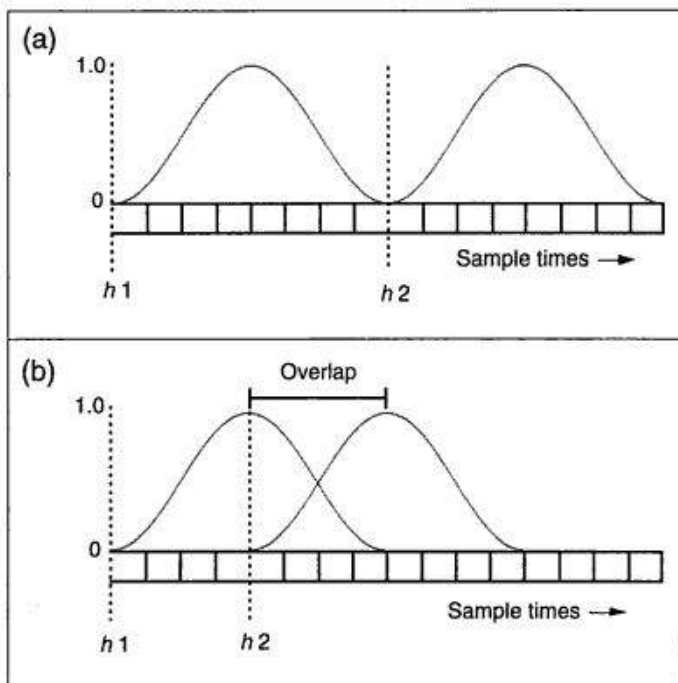
<http://eceserv0.ece.wisc.edu/~sethares/htmlRT/vocoders/phasevocoder.html>

**FFT Size and Zero-padding**

- FFT 사이즈의 선택은 하나의 소리에 적용할 계획 입력 변환에 따라 달라짐
- 하나의 프레임 사이즈가 128일때 FFT 사이즈는 256

**Hop Size**

- 짧은 hop size는 더 연속적인 윈도우 오버랩을 나타냄
- hop size는 프레임 사이즈의 일부분



**Figure 13.16** Varying hop size for analysis windows that are eight samples long.  $h_1$  and  $h_2$  are the starting times for each window. (a) Nonoverlapping windows when hop size = window size. (b) Overlapping windows when hop size is less than window size. In this case the hop size is four samples.

**Typical Parameter Values**

- PV의 정확한 파라미터 세팅 : 전통적인 악기 사운드들을 정확하게 재합성 가능
- PV의 최우선의 파라미터 세팅 규칙
  1. Frame size : 낮은 주파수 구간을 캡처하기에 적당해야 함  
 너무 작은 프레임 사이즈는 포먼트가 보존되어 있어도, 피치 변화
  2. Window type : rectangular를 제외한 표준적인 모든 타입 가능
  3. FFT size : frame size의 두배로
  4. Hop size : frame size의 여덟배  
 윈도우 타입의 -3dB 포인트에서 오버랩 됨

## Window closing

- 어떠한 윈도우 사이즈 세팅도 구간의 배음들에 대해 열갈린 분석 하게 됨
- 이를 위해, Window closing을 함
- : 높은 시간, 낮은 주파수 레졸루션 => 낮은 시간, 높은 주파수 레졸루션 점진적으로 작업
- 몇몇 STFT에서는 최적의 윈도우 사이즈를 찾기 위해, 이를 이용
- 분석된 사운드가 화성적인 구조를 가진다면, pitch-synchronous 분석이 잘 된다.

## Tracking Phase Vocoder

- 현재 많이 사용되는 PV : tracking phase vocoder
  - : 스펙트럼에서 가장 두드러진 peak를 따라 트래킹
- 주파수 재합성이 분석 윈도우의 배음들로 제한됨. 주파수의 변화에 따른다.
- Amp envelop과 frequency envelop 세트로 결과가 나옴
  - : 이를 가지고 사인파 오실레이터 बैं크를 조정
- STFT(같은 공간의 बैं크를 가짐)보다 정확한 분석 결과(다양한 공간을 가지므로)
- 장점 : Overlap-add frame보다 변환상 더 확고하다.
  - (앰프엔벨롭과 주파수엔벨롭 때문)
- 단점 : STFT와 달리 파라미터 셋팅에 따라 분석의 질이 틀리다

## Operation of the TPV

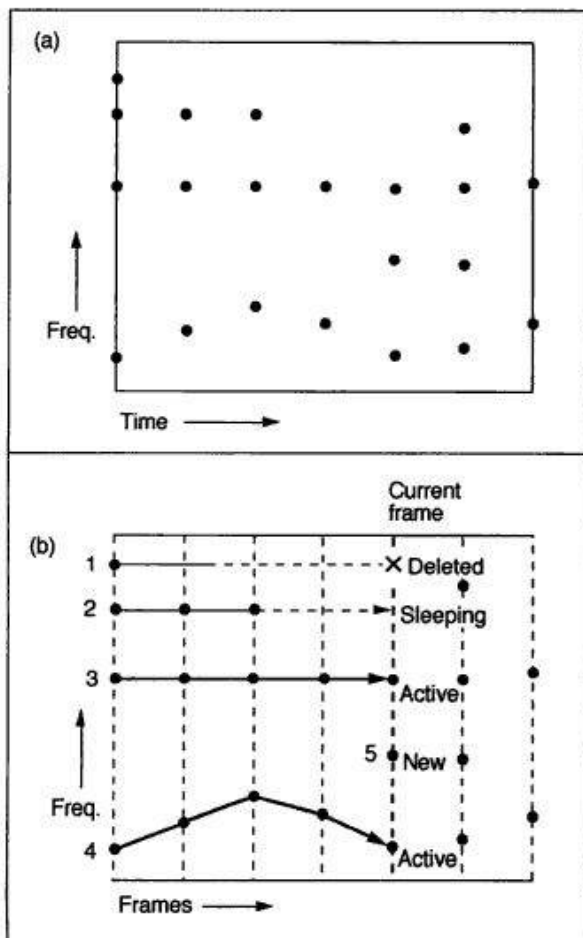
1. 사용자에게 의해 분류된 frame size / window type / FFT size / hop size를 사용하는 STFT 계산
2. dB에서 제공한 음량 스펙트럼 만듦
3. 스펙트럼에서 peak의 주파수 숫자 찾음
4. 각 주파수 피크의 음량, 위상 계산
5. 이전 프레임의 peak와 현재 프레임의 peak를 매치. 각 피크를 주파수 트랙으로 배당
6. 분석 파라미터로 모든 변형을 적용
7. additive 재합성이 필요하면, 각 주파수 트랙의 사인파를 생성하고, 이것을 다 더해 출력 각 사인파 요소들의 순간적인 음량, 위상 주파수는 삽입한 값들에 의해 계산 됨 (아니면, 이전에 언급한 다른 재합성 방법 사용)

## Peak Tracking

- tracking phase vocoder : 스펙트럼에서 가장 두드러진 주파수 궤도를 따름
- Tracking algorithm : 분석된 사운드의 타입 조정에 사용
  - 스피치, 하모닉 스펙트럼, 부드러운 인하모닉 스펙트럼, 노이즈 등
- guide로 트래킹하는 과정
  - ① peak identification
  - ② 시간에 따른 주파수 guide 셋트의 적용
  - ③ 현재 값에서 가장 peak가 가까운 주파수를 찾아 나감

- 방법

1. 현재 값과 매치하는 게 있으면, guide 지속
  2. guide가 지속할 수 없으면, "sleeping"
  3. guide가 약간의 지속 이후에 다시 나타나지 않으면, 삭제  
guide hysteresis  
: guide의 tracking을 지속. 분류된 음량 범위 아래로 약간 떨어짐  
반복적으로 fade하는 guide의 청각적 문제점을 완화하고,  
peak tracker에 의해 0점으로 자르고, 다시 fade in
  4. 만일 guide 사이에 충돌이 있으면, 가까운 guide로 가고, 충돌했던 다른 guide는 maximum peak deviation(최대 피크 편차)로 다른 guide를 찾는다. 주파수 대역은사용자에 의해 분류된다.
  5. 만일 현재 guide에 기대되는 피크가 있으면, 그 후에 새로운 guide를 시작한다.
- windowing 과정 : 일시적인 어택처럼 빠르게 움직이는 파형에서 특히 tracking의 정확성을 돕는다.
- 시간이 reverse 되면 : 날카로운 어택 사운드 프로세싱하는데, tracking algorithm을 돕게 됨 => 이를 바탕으로 배음 추적이 쉽다.



**Figure 13.17** Peak identification and tracking. (a) Isolation of a set of spectrum peaks. (b) Fitting frequency guides to peaks. Guide 1 at the top did not wake up after three frames, so it is deleted. Guide 2 is still sleeping. Guides 3 and 4 are active. Guide 5 starts from a new peak.

### Editing Analysis Envelopes

- 재합성된 파라미터를 바꿈으로써 소리의 변형을 생성
- 예) 재생 시 출력 사이즈를 변경하면 시간 확장과 압축 효과를 볼 수 있음
- 소음성 사운드 상에서 시간 확장이 실행될 때,  
개별적인 sine wave들은 살아나게 되고 소음성질 상실
- 복잡한 음악적인 변형을 생성해내기 위해서는 TPV에 의해 생성된 분석 데이터를 편집  
(주파수, 음량 그리고 위상 커브)  
( Moorer 1978; Dolson 1983; Gordon and Strawn 1985 ).

### Cross-synthesis with the Phase Vocoder

- 적은 에디팅으로 소리 변형시키는 다른 가능성으로 cross-synthesis 이 있다.
- Cross-synthesis는 하나의 테크닉이 아님 이것은 여러 가지의 형식(form)을 취하는 것
- 가장 일반적인 형식 하나의 스펙트럼에서 중요한 기능을 취하여 다른 스펙트럼의 주요 기능을 컨트롤 하는 것
- 사운드 A의 각각의 주파수 요소의 세기가 사운드 B의 상응하는 주파수 요소의 세기를 조정하는 것
- 이것은 스펙트럼 A에 있는 각각의 포인트들을 곱함으로써 스펙트럼 B에 상응하는 포인트들에 의해 실행
- Cross-synthesis의 이러한 타입을 위한 또 다른 용어로 filtering by convolution이 있다

### Computational Cost of the Phase Vocoder

- Phase Vocoder는 계산적으로 비용이 많이 드는 명령
- Tracking phase vocoder는 내부 중심이 효과적인 FFT 알고리즘을 사용하여 실행됨에도 불구하고 컴퓨터 파워의 많은 양 사용
- PV는 또한 많은 양의 분석 데이터를 생성  
어떤 경우 분석된 샘플 데이터의 사이즈보다 더 큰 경우도 있음
- 테크닉의 성장은 계산을 줄이고 공간을 보존하는 데에 적용
- 예) TPV에 의해 생성된 envelope은 낮은 샘플링 레이트에서 계산  
이러한 컨트롤 기능은 오디오의 샘플링 레이트보다 천천히 변화하는 경향이 있기 때문에 오디오의 질을 떨어뜨리지 않는다.

### Accuracy of Resynthesis

- 모든 Fourier-기반의 재합성에 있어 정확성은 분해절차의 분해도에 의해 제한
- 숫자의 반올림, windowing, peak-tracking, envelope 기능 샘플링에 의한 작은 에러
- 오차 또는 차이를 합성/재합성 에러로 간주  
재합성된, 시그널의 "clean" 파트와 같은 quasi-harmonic 분배와 시그널의  
"dirty" 파트와 같은 소음 요소를 언급하는 것은 일반적
- 몇몇 PV들은 앰플리튜드와 주파수 데이터만 보존. phase 정보를 무시하는 옵션을 가지고 있음

- 이 결과, 데이터 감소와 계산시간상의 이에 상응하는 보존을 얻을 수 있으나, 재합성의 정확도 저하(예) 재합성된 waveform은 적당한 phase 데이터 없이는 동일한 기본 주파수 요소를 가지고 있음에도 불구하고 오리지널을 닮을 수 없다(Serra 1989).
- 일정한 지속적인 상태의 사운드의 경우, phase의 재정렬로 인하여 들을 수 없게 될 수도 있다. 그러나 transient와 quasi-steady-state tone의 높은-적합도를 위하여, phase 데이터는 적합한 명령으로 그들의 짧게-움직이고 변화하는 요소들을 재조합하므로 가치 있는 것으로 볼 수 있다.

### Problem Sounds

- PV는 harmonic적이고, 지속적이거나, 혹은 서서히 변하는 tone을 가장 잘 처리
- 사운드의 타임스케일 확대와 압축과 같은 변형의 결과로 자연스러운 사운드 효과를 얻을 수 있다.
- 본질적으로는 PV 테크닉으로 변형하기는 어렵다. 그것들에는, 귀에 거슬리거나 혹은 숨소리 같은 목소리, 모터, 몇 밀리초의 타임스케일 상에서 빠르게 변화하는 어떠한 사운드들, 그리고 공간의 소음을 가지고 있는 이 같은 소음성의 사운드들이 포함된다. 이러한 타입의 사운드들을 변형시킬 경우, 그 결과로 echo, flutter, 의도하지 않은 공명, 그리고 원치 않은 음색의 리버브 효과 등을 얻게 된다. 이것들은 주로 phase distortion 때문에 일어나며 분석된 데이터가 변형될 때 발생한다.

### Analysis of Inharmonic and Noisy Sounds

- Tracking Phase Vocoder로 여러 가지 인하모닉 사운드와 유폴타악기를 분석하고 재합성 할 수 있다는 것을 시연에 의하여 입증
- 이것은 소음성 시그널과 인하모닉 시그널을 주기적인 사인곡선 기능의 조합 (TPV가 Fourier 분석에 기초)
- 소음성 시그널은 저장되고 계산되는 시점으로부터 소비적으로(비경제적으로) 처리
- TPV의 경우 오리지널 사운드 샘플 바이트의 열 배가 넘기도 함

#### 참고문헌 및 사이트

Curtis Roads, *The Computer Music Tutorial*, pp. 566~577

<http://eceserv0.ece.wisc.edu/~sethars/htmlRT/vocoders/phasevocoder.html>

#### 작성자

한기열 (박사과정, 7기)